CLAIMS

What is claimed is:

1. A method comprising:

recording a first last use of a first canonical register in a block of code after a renaming, the first canonical register being mapped to a first original register; and

applying one of a first rollback and a first recovery to the first original register based on whether the recorded first last use occurs before a first last definition of the first original register in the block of code.

2. The method of claim 1 wherein applying one of the first rollback and the first recovery comprises:

applying the first rollback to the first original register if the recorded first last use occurs before the first last definition of the first original register; and

applying the first recovery to the first original register if the recorded first last use does not occur before the first last definition of the first original register.

3. The method of claim 2 wherein applying the first rollback comprises:

replacing a first reference to a first target register with the first canonical register when the first reference is a destination of a first last write to the first target register, the first target register corresponding to the first original register after the renaming; and

replacing a second reference to the first target register with the first canonical register when the second reference is a source of a first operation after the first last write to the first target register.

4. The method of claim 2 wherein applying the first recovery comprises:

copying the first target register to the first canonical register at end of the block.

5. The method of claim 4 wherein copying the first target register comprises:

copying the first target register to a first unused temporary register; and

copying the first unused temporary register to the first canonical register.

6. The method of claim 5 further comprises:

recording a second last use of a second canonical register in the block of code after the renaming, the second canonical register being mapped to a second original register; and

applying one of a second rollback and a second recovery to the second original register based on whether the recorded second last use of the second canonical register occurs before a second last definition of the second original register in the block of code.

7. The method of claim 6 wherein applying one of the second rollback and the second recovery comprises:

applying the second rollback to the second original register if the recorded second last use occurs before the second last definition of the second original register; and

applying the second recovery to the second original register if the recorded second last use does not occur before the second last definition of the second original register.

8. The method of claim 7 wherein applying the second rollback comprises:

replacing a third reference to a second target register with the second canonical register when the third reference is a destination of a second last write to the second target register, the second target register corresponding to the second original register after the renaming; and

replacing a fourth reference to the second target register with the second canonical register when the fourth reference is a source of a second operation after the second last write to the second target register.

9. The method of claim 8 wherein applying the second recovery comprises:

copying the second target register to the second canonical register at end of the block.

10. The method of claim 9 wherein copying the second target register comprises:

copying the second target register to a second unused temporary register before copying the first unused temporary register to the first canonical register; and

copying the second unused temporary register to the second canonical register.

11. A computer program product comprising:

a machine useable medium having program code embedded therein, the program code comprising:

computer readable program code to record a first last use of a first canonical register in a block of code after a renaming, the first canonical register being mapped to a first original register; and

computer readable program code to apply one of a first rollback and a first recovery to the first original register based on whether the recorded first last use occurs before a first last definition of the first original register in the block of code.

12. The computer program product of claim 11 wherein the computer readable program code to apply one of the first rollback and the first recovery comprises:

computer readable program code to apply the first rollback to the first original register if the recorded first last use occurs before the first last definition of the first original register; and

computer readable program code to apply the first recovery to the first original register if the recorded first last use does not occur before the first last definition of the first original register.

13. The computer program product of claim 12 wherein the computer readable program code to apply the first rollback comprises:

computer readable program code to replace a first reference to a first target register with the first canonical register when the first reference is a destination of a first last write to the first target register, the first target register corresponding to the first original register after the renaming; and

computer readable program code to replace a second reference to the first target register with the first canonical register when the second reference is a source of a first operation after the first last write to the first target register.

14. The computer program product of claim 12 wherein the computer readable program code to apply the first recovery comprises:

computer readable program code to copy the first target register to the first canonical register at end of the block.

15. The computer program product of claim 14 wherein the computer readable program code to copy the first target register comprises:

computer readable program code to copy the first target register to a first unused temporary register; and

computer readable program code to copy the first unused temporary register to the first canonical register.

16. The computer program product of claim 15 further comprises:

computer readable program code to record a second last use of a second canonical register in the block of code after the renaming, the second canonical register being mapped to a second original register; and

computer readable program code to apply one of a second rollback and a second recovery to the second original register based on whether the recorded second last use of the second canonical register occurs before a second last definition of the second original register in the block of code.

17. The computer program product of claim 16 wherein the computer readable program code to apply one of the second rollback and the second recovery comprises:

computer readable program code to apply the second rollback to the second original register if the recorded second last use occurs before the second last definition of the second original register; and

computer readable program code to apply the second recovery to the second original register if the recorded second last use does not occur before the second last definition of the second original register.

18. The computer program product of claim 17 wherein the computer readable program code to apply the second rollback comprises:

computer readable program code to replace a third reference to a second target register with the second canonical register when the third reference is a destination of a

second last write to the second target register, the second target register corresponding to the second original register after the renaming; and

computer readable program code to replace a fourth reference to the second target register with the second canonical register when the fourth reference is a source of a second operation after the second last write to the second target register.

19. The computer program product of claim 18 wherein the computer readable program code to apply the second recovery comprises:

computer readable program code to copy the second target register to the second canonical register at end of the block.

20. The computer program product of claim 19 wherein the computer readable program code to copy the second target register comprises:

computer readable program code to copy the second target register to a second unused temporary register before copying the first unused temporary register to the first canonical register; and

computer readable program code to copy the second unused temporary register to the second canonical register.

21. A system comprising:

a processor; and

a memory coupled to the processor to store program code, the program code, when executed, causing the processor to:

record a first last use of a first canonical register in a block of code after a renaming, the first canonical register being mapped to a first original register; and

apply one of a first rollback and a first recovery to the first original register based on whether the recorded first last use occurs before a first last definition of the first original register in the block of code.

22. The system of claim 21 wherein the program code causing the processor to apply one of the first rollback and the first recovery causes the processor to:

apply the first rollback to the first original register if the recorded first last use occurs before the first last definition of the first original register; and

apply the first recovery to the first original register if the recorded first last use does not occur before the first last definition of the first original register.

23. The system of claim 22 wherein the program code causing the processor to apply the first rollback causes the processor to:

replace a first reference to a first target register with the first canonical register when the first reference is a destination of a first last write to the first target register, the first target register corresponding to the first original register after the renaming; and

replace a second reference to the first target register with the first canonical register when the second reference is a source of a first operation after the first last write to the first target register.

24. The system of claim 22 wherein the program code causing the processor to apply the first recovery causes the processor to:

copy the first target register to the first canonical register at end of the block.

25. The system of claim 24 wherein the program code causing the processor to copy the first target register causes the processor to:

copy the first target register to a first unused temporary register; and

copy the first unused temporary register to the first canonical register.

26. The system of claim 25 wherein the program code further causes the processor to:

record a second last use of a second canonical register in the block of code after the renaming, the second canonical register being mapped to a second original register; and

apply one of a second rollback and a second recovery to the second original register based on whether the recorded second last use of the second canonical register occurs before a second last definition of the second original register in the block of code.

27. The system of claim 26 wherein the program code causing the processor to apply one of the second rollback and the second recovery causes the processor to:

apply the second rollback to the second original register if the recorded second last use occurs before the second last definition of the second original register; and

apply the second recovery to the second original register if the recorded second last use does not occur before the second last definition of the second original register.

28. The system of claim 27 wherein the program code causing the processor to apply the second rollback causes the processor to:

replace a third reference to a second target register with the second canonical register when the third reference is a destination of a second last write to the second target register, the second target register corresponding to the second original register after the renaming; and

replace a fourth reference to the second target register with the second canonical register when the fourth reference is a source of a second operation after the second last write to the second target register.

29. The system of claim 28 wherein the program code causing the processor to apply the second recovery causes the processor to:

copy the second target register to the second canonical register at end of the block.

30. The system of claim 29 wherein the program code causing the processor to copy the second target register causes the processor to:

copy the second target register to a second unused temporary register before copying the first unused temporary register to the first canonical register; and

copy the second unused temporary register to the second canonical register.